Enabling Distributed Event Management: Interoperability for Automated Response and Prevention

Sean Barnum
George Saylor
Aug 2011
The HS SEDI FFRDC is managed and operated by The MITRE Corporation for DHS.
Premise

- Supporting attack detection and automated courses of action in today’s advanced threat environment requires integration and correlation of diverse sets of event producers as well as standardized structured event representations to support interoperability.

Solution Resources
- Event Management Automation Protocol (EMAP)
- Cyber Observable eXpression (CybOX)
The Problem

- Lots of events, all in different formats
The Problem (Continued)
What is EMAP?

• A program to establish a protocol to enable standardized content, representation, exchange, correlation, searching, storing, prioritization, and auditing of event records within an organizational IT environment
• A specification to meet these goals and define how relevant technical specifications will interact to achieve the desired outcomes.
• Develop and implement validation of products that conform to the standard
What is EMAP? (Continued)

**Languages**
- Express logs and policies
  - Common Log formats
  - Log correlation rules
  - Logging configuration
  - Audit Settings
  - Normalization
  - Incident Description
  - Observable Description

**Metrics**
- Event scoring framework
  - Severity of logged events
  - Criticality
  - Impact

**Enumerations**
- Convention for identifying and naming
  - Log taxonomy
  - Enrichment information
  - Observables

(U)
Event Funnel

# of events
millions

- firewalls
- intrusion detection
- antivirus/malware
- vulnerability scans
- system/application

enumerations

normalization

languages

correlation

metrology

prioritized threats

tens
	hundreds

thousands

millions
How it all works

PARSE \rightarrow FILTER \rightarrow CORRELATE
Notional EMAP Components

- **Common Event Expression (CEE)**
  - A suite of specifications to define taxonomy, syntax, transport, logging recommendations, and parsing information about event records

- **Open Event Expression Language (OEEL)**
  - A language to express parsing and normalization logic using CEE Profiles to convert event records into CEE

- **Common Event Rule Expression (CERE)**
  - A common format to express rules for pattern matching, filtering, and correlation
Notional EMAP Components

• Common Event Scoring System (CESS)
  – A specification that provides metrics of event severity and impact based on multiple factors

• Cyber Observable eXpression (CybOX)
  – A language to express cyber observable events or stateful measures that provides a common foundation for many of the other standards
CEE Components

- CEE Dictionary
- CEE Taxonomy (CEET)
- Common Event Log Recommendations (CELR)
- Common Log Syntax (CLS)
- Common Log Transport (CLT)
CEE Architecture
OEEL

• Provide a standardized ability to represent parsing logic external to the parsing application
  – Provide vendors and consumers to express and share parsing logic in a standard format
  – Simplify product development
  – A way to change a native log into a standard format (example Apache to CEE)
  – Combine multiple log and data sources together into common output
CERE

• Provide vendors and consumers a way to express and share rules for pattern matching, correlation, and filtering of logs
  – Support distributed multi-vendor enterprises
  – Aid in acquisition
  – Simplify sharing detection rules to public
  – Achieve this with minimal impact to vendors and consumers
CESS

• Provide a metric for scoring the severity, criticality, and impact of a given event, set of events, or incident
  – Leverage EMAP components to derive reasonably accurate scoring
  – Use CESS to assist in filtering and correlation
  – Provide vendors, responders, analysts a reliable and repeatable means of calculating and sharing this information
Enabling Automated Response and Prevention

• Interoperability
  – Mapping observables and indicators to detection rules
  – Mapping attack patterns to observables and detection rules
  – Mapping malware characteristics to observables and rules
  – Incident Management automation through information exchange of standardized content
The Cyber Observables construct is intended to capture and characterize events or properties that are observable in the operational domain.

These observable events or properties can be defined in rules or used to adorn the appropriate portions of the attack patterns in order to tie the logical pattern constructs to real-world evidence of their occurrence or presence for attack detection and characterization.

By capturing them in a structured fashion, the intent is to enable future potential for detailed automatable mapping and analysis heuristics.
Cyber observables apply to numerous domains

- Detailed attack patterns
- Malware characterization
- Operational Events
- Logging
- Cyber situational awareness
- Incident response
- Forensics
- Etc.
A Brief History of Cyber Observables

- September 2009: Concept introduced to CAPEC in Version 1.4 as future envisioned adornment to the structured Attack Execution Flow
- June 2010: Broader relevance to MSM recognized leading to CAPEC, MAEC & CEE teams collaborating to define one common structure to serve the common needs
- August 2010: Discussed with US-CERT at GFIRST 2010
- December 2010: Cyber Observables schema draft v0.4 completed
- December 2010: Discussions with Mandiant for collaboration and alignment between Cyber Observables and Mandiant OpenIOC
- January 2011: Discussed & briefed with MITRE CSOC
- February 2011: Discussed & briefed with NIST – EMAP and US-CERT who also have a need for this construct and had begun to work on parallel solutions
- May 2011: Schematic alignment and integration with CEE
Common Cyber Observables (CybOX) Schema
Common Cyber Observables (CybOX) Schema
Common Cyber Observables (CybOX) Schema

Observable
  Measure Source
  Measure
  Stateful_Measure
  Event
  Action

Object
Object State
 Defined Object
 Custom Attributes

File System
 GUI
 IPC
 Internet
 Module
 Registry
 Memory
 Network
 Daemon

CAPEC
MAEC
CEE
### Various Defined Object Schemas

- Account
- Disk
- Disk Partition
- DNS Cache
- Email Message
- File
- GUI
- Library
- Package
- Memory
- Network Connection
- Network Route
- Linux Package
- Product
- Service
- Socket
- System
- User Session
- Volume
- Win Critical Section
- Win Driver
- Win Event
- Win Event Log
- Win Kernel
- Win Kernel Hook
- Win Handle
- Win Mailslot
- Win Mutex
- Win Named Pipe
- Win Network Route
- Win Prefetch
- Win Registry
- Win Semaphore
- Win System Restore
- Win Task
- Win Thread
- Win Waitable Timer
- X509 Certificate

… (more on the way)
Cyber Observable Use Cases

- Potential ability to analyze data from all types of tools and all vendors
- Improved sharing among all cyber observable stakeholders
- Detect malicious activity from attack patterns
- Empower & guide incident management
- Identify new attack patterns
- Prioritize existing attack patterns based on tactical reality
- Ability to metatag cyber observables for implicit sharing controls
- Enable automated signature rule generation
- Enable new levels of meta-analysis on operational cyber observables
- Potential ability to automatically apply mitigations specified in attack patterns
- Etc....
Use Case: Detect Malicious Activity

Current:
- Manual effort to pull together data across many sensors
  - Results in limited situational awareness
- Attack patterns and rules are typically too detailed (physical signatures) or ambiguous prose
- High level of effort
- High false negatives & positives

CybOX-enabled:
- Diverse set of sensors output data in common format
- Attack patterns and rules can be defined in a uniform fashion
- Pattern matching and analysis heuristics can be easily automated
Use Case: Incident Response Data Capture

- **Current:**
  - Very manual
  - Inconsistent between analysts & organizations
  - Prose-based and imprecise
  - Difficult to automate capture and actionable alerts

- **CybOX-enabled:**
  - Improved consistency
  - Ability to tie everything together
  - Simplified and automated data capture
  - Alerts become actionable for automation
Use Case: Malware Analysis

- Current:
  - Difficult to combine different analysis perspectives or tools
  - Difficult to share info
  - Difficult to recognize if malware has been seen before
  - Does not scale well

- CybOX(MAEC)-enabled:
  - Easier to integrate different forms of analysis, different tools and even information from different sources
  - Easier to share information
  - Easier to recognize malware (including variants and perturbations)
  - Enables automated interaction among the various dimensions of malware analysis
Use Case: Malware Artifact Hunting

■ Current:
  – Very manual
  – Often imprecise and inconsistent
  – Localized

■ CybOX(MAEC)-enabled:
  – Very automated
  – Consistent
  – Enables broad, non-localized sharing and hunting
Use Case: Host Based Detection

Dynamic Analysis Engine
- Anubis
- CWSandbox
- ThreatExpert
- Etc.

Malware Binary

Engine Output

Sandbox -> MAEC Translator

MAEC

Host-based Scanner

OVAL

© 2011 The MITRE Corporation. All rights reserved.
Use Case: IR/IM Alerts

■ Current:
  – Typically unstructured prose
  – Labor intensive and slow
  – Limited actionable (in an automated fashion) data

■ CybOX-enabled:
  – Structured and consistent
  – Alert generation can be much faster and less labor intensive
  – Potentially actionable in an automated context
Notional Flow of a Modern Security Incident

1. An attack on an information system occurs involving social engineering, vulnerability exploit, malware + command and control (C2).

2. CybOX-enabled operational sensors (IDS, host-based, etc.) pick up anomalous activity and report it in CEE/CybOX formats.

3. Automated analysis tools & rules attempt to match anomalous activity against CybOX-adorned CAPEC attack patterns but discover no matching patterns.

4. Incident is reported – Incident Response/Management process is initiated.

5. IR personnel capture discovered detail of incident in CybOX-compliant formats, including CEE.

6. IR personnel detect malware as part of the ongoing attack.
7. Malware undergoes automated analysis (dynamic and/or static) and results are captured in MAEC (CybOX-integrated) language.

8. Malware analysts are able to correlate the current malware instance with a broad range of pre-existing malware samples and analysis data from MAEC-enabled repositories and zoos.

9. Malware analysts capture new discovered detail of the malware in MAEC format, including the CWE or CVE exploited.

10. Sample and analysis data from current malware instance are entered into appropriate malware repositories and zoos.

11. CybOX observables of malware effects on hosts are extracted from MAEC content to generate OVAL checks to determine if any given host has been infected/affected by the current malware instance.

12. OVAL checks are distributed and run against other areas of the domain or enterprise to determine breadth of compromise.
Notional Flow of a Modern Security Incident (cont.)

13. IR/IM personnel apply appropriate mitigations/remediations to negate the effects of the attack.

14. A new CAPEC attack pattern is authored to describe this new observed attack behavior, and is adorned as appropriate with CybOX content observed for this pattern in the operational space.

15. IR/IM personnel issue relevant alerts for the observed incident including the new CAPEC pattern, MAEC bundle and related CEE/CybOX content.

16. Secure development takes advantage of this new CAPEC pattern to: define/refine appropriate security policy, training & requirements; guide security engineering (control selection), architectural risk analysis, secure code review and security testing; identify relevant CWE weaknesses, CVE vulnerabilities & CCE configuration issues; prioritize relevant CAPEC patterns based on real-world observed prevalence/frequency profiled through automated observation of CybOX patterns in the operational space.
Where is CybOX today?

- Currently integrated into CAPEC
- Currently integrated into MAEC
- In process of being integrated into CEE
- Part of the strategic approach for EMAP
- Part of the strategic vision for IR/IM with US-CERT
- Continued integration discussions planned for Mandiant OpenIOC once initial drafts of Object schemas are complete
- Currently being evaluated for integration into multiple research projects
- Website should be up soon
Questions / Comments?

Sean Barnum
sbarnum@mitre.org

George Saylor
george.saylor@nist.gov

EMAP Developer Days (Aug 29-30) – NIST
http://scap.nist.gov/events

ITSAC (Oct 31 - Nov 2) – Crystal City